A Dynamic Lot-Sizing Model with Demand Time Windows

Econometric Institute Report EI-9948/A

Chung-Yee Lee*
Sila Çetinkaya *
Department of Industrial Engineering
Texas A&M University
College Station, TX 77843-3131

Albert P.M. Wagelmans
Econometric Institute and RIBES
Erasmus University Rotterdam
P.O. Box 1738
3000 DR Rotterdam
The Netherlands

September 1998
Revised: November 1999

* This research was supported in part by NSF Grant **DMI-9908221**.

Key Words: lot-sizing, dynamic programming, time windows

Abstract:

One of the basic assumptions of the classical dynamic lot-sizing model is that the aggregate demand of a given period must be satisfied in that period. Under this assumption, if backlogging is not allowed then the demand of a given period cannot be delivered *earlier* or *later* than the period. If backlogging is allowed, the demand of a given period cannot be delivered *earlier* than the period, but can be delivered later at the expense of a backordering cost. Like most mathematical models, the classical dynamic lot-sizing model is a simplified paraphrase of what might actually happen in real life. In most real life applications, the customer offers a grace period - we call it a *demand time window* - during which a particular demand can be satisfied with no penalty. That is, in association with each demand, the customer specifies an earliest and a latest delivery time. The time interval characterized by the earliest and latest delivery dates of a demand represents the corresponding time window.

This paper studies the dynamic lot-sizing problem with demand time windows and provides polynomial time algorithms for computing its solution. If shortages are not allowed, the complexity of the proposed algorithm is $O(T^2)$. When backlogging is allowed, the complexity of the proposed algorithm is $O(T^3)$.

1. Problem Context and Definition

The classical dynamic lot-sizing problem considers a facility, possibly a warehouse or a retailer, which faces dynamic demand for a single item over a finite horizon (Wagner and Whitin 1958). The facility places orders for the item from a supply agency, e.g., a manufacturer or a supplier, which is assumed to have an unlimited quantity of the product. The model assumes a fixed ordering (setup) cost, a linear procurement cost for each unit purchased, and a linear holding cost for each unit held in inventory per unit time. Shortages at the warehouse/retailer may or may not be allowed, and depending on how shortages are modeled, a linear stockout cost may accrue for every unit backordered per unit time. Given the time varying demand and cost parameters, the problem is to decide when and how much to order at the facility in each period so that all demand is satisfied at minimum cost.

The basic assumption of the classical dynamic lot-sizing model is that the time varying demand is known in advance. Let T denote the length of the planning horizon over which the demands, denoted by d_i, $i \in \{1,\ldots,T\}$, should be satisfied. Under the assumptions of the classical model, d_i represents the aggregate demand (placed by all customers) that *must be* satisfied in period $i \in \{1,\ldots,T\}$. If backlogging is not allowed then d_i cannot be delivered *earlier* or *later* than $i \in \{1,\ldots,T\}$. If backlogging is allowed, d_i cannot be delivered *earlier* than i, but it can be delivered later at the expense of backordering costs.

Like most mathematical models, the classical dynamic lot-sizing model is a simplified paraphrase of what might actually happen in real life. The assumption that the values of d_i, $i \in \{1,\ldots,T\}$, are known in advance is applicable if supply contracts are signed ahead of time designating deliveries for the next few periods (Bramel and Simchi-Levi 1997, p. 165). However, under a typical supply contract, the customer offers a grace period - we call it a *demand time window* - during which a particular demand can be satisfied with no penalty. That is, associated with each d_i, the customer specifies an earliest and a latest delivery time, denoted by E_i and L_i, respectively, where $E_i \leq L_i$. Hence, the interval $[E_i, L_i]$ represents the time window corresponding to d_i.

As the title suggests, this paper studies the dynamic lot-sizing problem with demand time windows and provides polynomial time algorithms for computing its solution. The following two cases are considered:

- Shortages are not allowed so that each d_i *must be* delivered during its corresponding time window, i.e., no earlier than E_i and no later than L_i.
- Backlogging is allowed, i.e., demand d_i cannot be delivered *earlier* than E_i, but can be delivered later than L_i at the expense of backordering costs.

The dynamic lot-sizing problem with demand time windows has important applications in third party warehousing and vendor managed inventory practices. A detailed discussion of practical motivations and a brief review of relevant literature are presented in Section 2. The notation is introduced and some structural properties of the problem are proved in Section 3. Sections 4 and 5 focus on developing polynomial time algorithms for computing the optimal solution under different backordering assumptions. If shortages are not allowed, the complexity of the proposed algorithm is $O(T^2)$, and if backlogging is allowed, the complexity of the proposed algorithm is $O(T^3)$. Section 5 presents an interesting application in location planning which is related to the dynamic lot-sizing model with demand time windows. Finally, a brief summary and our concluding remarks are furnished in Section 6.

2. Problem Motivations and Related Literature

The dynamic lot-sizing problem has received a significant amount of academic attention since it was first introduced more than four decades ago. The solution technique, known as the Wagner-Whitin algorithm, has long been regarded as one of the basic methods in production planning and inventory control. For a brief summary of results and the history of this model, see the textbooks by Bramel and Simchi-Levi (1997), Johnson and Montgomery (1974), and Silver et al., (1996). In the early years, due to the limitations of computational technology, the Wagner-Whitin algorithm seemed to require excessive computational and storage requirements. Hence, several heuristic procedures and efficient implementation techniques have been investigated (e.g., see Evans 1985, and Silver and Meal 1973).

In the last decade, three important papers, Aggarwal and Park (1993), Federgruen and Tzur (1991), and Wagelmans et al. (1992), improved the time complexity for obtaining an optimal solution from $O(T^2)$ to $O(T\log T)$ for general problems, and to $O(T)$ for problems with a special cost structure. For interesting generalizations of Wagner and Whitin's model, see Bitran et al. (1984), Chen et al. (1994), Chen and Lee (1995), Chung and Lin (1988), Florian et al. (1980), Van Hoesel and Wagelmans (1996, 1997), Lee (1989), Lee and Denardo (1986), Maes and Van Wassenhove (1985), Swoveland (1975), and Zangwill (1966).

The classical dynamic lot-sizing model is widely applicable in certain settings. However, the more general problem with demand time windows also has a significant practical importance. Our observations in real life applications attest that associated with each demand, the customer allows a grace period for timely delivery. This grace period is called *a demand time window* in this paper.

In recent years, time definite delivery (TDD) agreements have become a popular component of supply contracts in third party warehousing (Çetinkaya and Lee 1999). Under a TDD agreement, the inventory information at the downstream supply chain member, e.g., the customer, is accessible to the supplier, e.g., the warehouse. Reviewing the customer's inventory levels, the supplier makes decisions regarding the quantity and timing of re-supply, but assures timely delivery via imposing a maximum holding time (i.e., a time window) for each delivery. In a representative practical situation, a third party logistics company provides warehousing and transportation for a manufacturer and guarantees TDD for outbound deliveries to the customers. Such an arrangement is particularly advantageous for effective vendor managed inventory where the vendor assumes responsibility for managing inventories at retailers using advanced data retrieval systems (Aviv and Federgruen 1998, Çetinkaya and Lee 1998, Parker 1996), and guarantees timely delivery via satisfying demand time window constraints specified by the retailers.

The concept of a demand time window has received attention in vehicle routing research. Solomon and Desrosiers (1988) provide a detailed survey of the literature in this area. For other recent papers in vehicle routing under *delivery due date constraints*, see Bramel and Simchi-Levi (1997, Chapter 7), Desrochers et al. (1992), Kohl and Madsen (1997), Fisher et al. (1997), and Kohl et al. (1998).

The concept of a demand time window has also been analyzed in the context of job shop scheduling. Anger et al. (1986) study a single machine problem where the objective is to maximize the number of jobs finished within a production due date window. Other recent studies in scheduling of *production due date constraints* include Cheng (1988), Kramer and Lee (1993, 1994), Liman and Ranaswamy (1994), Liman et al. (1996), Weng and Venture (1994), and Venture and Weng (1996).

Although the concept of a demand time window has been analyzed in relation to *delivery due dates in vehicle routing* and *production due dates in scheduling*, the problem of interest in this paper has not been addressed in the previous literature. This paper focuses on a generalization of the dynamic lot-sizing model where each demand is characterized by a time window. That is, given the time varying demands characterized by the earliest and latest due dates, the problem is about *when* and *how much* to order in each period so that demands can be satisfied during their corresponding time windows with a minimal cost.

In the next section we present our notation, and provide a mathematical programming formulation of the dynamic lot-sizing problem with demand time windows. This formulation is helpful in understanding the difference between the classical problem and the more general problem with demand time windows. In the following section, we also present some important structural properties of the problem with demand time windows. These structural properties are useful for developing polynomial time algorithms as we discuss later in the paper.

3. Notation and an Integer Programming Formulation of the Problem

In the classical dynamic lot-sizing model, demands that are due in a given period can be aggregated and these aggregated demands can be indexed according to the time period in which they are due. Consequently, the number of demands (aggregated according to their due dates) during the planning horizon cannot exceed T. For the more general problem with demand time windows, two demands with different latest due dates cannot be aggregated, even if their earliest due dates are the same, and vice versa. In this case, we can only aggregate those demands with the same earliest and latest due dates. After aggregation, the number of demands, denoted by n, is restricted by the maximum number of distinct demand time windows, i.e., the number of distinct $[E_i, L_i]$ intervals. Therefore, for the more general case with time windows $n \le T(T\text{-}1)$, whereas for the classical problem $n \le T$. Considering this basic difference between the classical problem and the more general problem with demand time windows, let us summarize the notation that has been in use and introduce some additional notation.

Let t and T denote the period index and the length of planning horizon, respectively. It follows that $t \in \{1,\ldots,T\}$. Recall that n is the number of demands throughout T. For each $i \in \{1,\ldots,n\}$, d_i represents the quantity of demand i; E_i denotes the earliest due date of demand i; and L_i denotes the latest due date of demand i. For each $t \in \{1,\ldots,T\}$, let us also define the following:

- I_t^+ denotes the on-hand inventory at the end of period t.
- I_t^- denotes the amount backordered at the end of period t.
- K_t denotes the fixed cost of ordering/procurement (setup cost) in period t.
- p_t denotes the unit ordering/procurement cost in t.
- h_t denotes the unit holding cost in t.
- b_t denotes the unit backlogging cost in t.

Under the cost assumptions of the problem under consideration, we have the following:

- A holding cost is accrued if inventory is replenished earlier than E_i for satisfying d_i.
- A backlogging cost is accrued if inventory is replenished later than L_i for satisfying d_i.
- No additional cost (other than ordering/procurement costs) is incurred, if d_i is ordered and satisfied during $[E_i, L_i]$.

In order to model the effects of the above observations in a mathematical programming formulation, we define the following variables:

- x_t denotes the amount produced in period t.
- $y_t = 1$ if $x_t > 0$, and 0 otherwise.
- If $t \in \{ E_i,\ldots,L_i \}$then d_{it} is the amount of d_i that is scheduled to be satisfied in period t; otherwise $d_{it} = 0$.
- M is a very large number.

Using this notation, the dynamic lot-sizing problem with demand time windows can be formulated as an integer programming (IP) problem as follows:

$$Min \sum_{t=1}^{T} (p_t x_t + K_t y_t + h_t I_t^{+} + b_t I_t^{-}) \qquad (1)$$

Subject to

$$x_t + (I_{t-1}^{+} - I_{t-1}^{-}) - \sum_{i=1}^{n} d_{it} = (I_t^{+} - I_t^{-}) \qquad t = 1,\ldots,T, \qquad (2)$$

$$x_t \leq M\, y_t \qquad t = 1,\ldots,T, \qquad (3)$$

$$\sum_{t=E_i}^{L_i} d_{it} = d_i \qquad i = 1,\ldots,n, \qquad (4)$$

$$d_{it} \geq 0 \qquad i = 1,\ldots,n;\ \mathrm{t} = E_i,\ldots,L_i \qquad (5)$$

$$d_{it} = 0 \qquad i = 1,\ldots,n;\ \mathrm{t} = 1_i,\ldots,E_i\text{-}1 \qquad (6)$$

$$d_{it} = 0 \qquad i = 1,\ldots,n;\ \mathrm{t} = L_i+1,\ldots,T_i \qquad (7)$$

$$x_t \geq 0, I_t^+ \geq 0,\ I_t^- \geq 0, y_t \in \{0,1\} \qquad t = 1,\ldots,T, \tag{8}$$

$$I_0^+ = I_0^- = 0 \tag{9}$$

It is worth noting that, in the above IP model, d_{it} is a decision variable while d_i is a given parameter. Naturally, the above formulation is an extension of the classical dynamic lot-sizing model. Observe that equations (4)-(7) are additional constraints representing the demand time window considerations. If E_i = L_i for all i, then our IP model reduces to the classical dynamic lot-sizing model.

The above IP formulation is useful in comparing the computational challenges of the more general problem with demand time windows with the classical dynamic lot-sizing problem. However, before we provide a discussion of this issue, let us analyze the optimality properties of our problem and present a polynomial algorithm for the case where backordering is not allowed. We will revisit the IP formulation shortly before we conclude our discussion about the time complexity of the proposed algorithm for the case where backordering is not allowed.

4. Optimality Properties

For the sake of simplicity, we momentarily assume that the unit ordering/procurement cost is stationary, i.e., $p_t = p$ for all t**.** Later, in Section 7, we will generalize our results to the case where there are no speculative motives to hold inventory or to backorder. Under the assumption that the unit ordering/procurement cost is stationary, the total ordering/procurement cost (not including the fixed cost of ordering, i.e., the setup cost) can be ignored for the purpose of cost minimization. In this case, the following optimality properties are satisfied.

Property 1: Regardless that backordering is allowed or not, there exists an optimal solution in which demand is not split. That is, there exists an optimal policy such that, for each demand the entire quantity is covered by the same replenishment order.

Proof: This property is true since there is no limit on the size of the replenishment quantity, i.e., there are no capacity constraints. Using this observation, it is easy to show that any optimal solution (in which demand is split) can be transformed into an equivalent solution in which demand is not split.

Property 2: Suppose that $t_1 < t_2 < \ldots < t_r$ are the successive replenishment periods in an optimal solution. Then the following observations are true.

i) If a replenishment period lies within the time window of a demand, then no holding or backlogging costs are incurred for satisfying that demand.

ii) If $t_{k-1} < E_i \leq L_i < t_k$, for some $k = 1,2,\ldots,r$-1, then d_i is satisfied by using the replenishment received in either t_{k-1} or t_k depending on which of the following is cheaper:

- to replenish in t_{k-1} and carry d_i units of inventory until period E_i, or
- to replenish in t_k and backlog d_i units of demand until period L_i.

iii) If $E_i > t_r$, then d_i is satisfied using the replenishment order received in t_r (in this case a holding cost is incurred).

iv) If $L_i < t_1$, then d_i is satisfied using the replenishment order received in t_1 (in this case a backordering cost is incurred).

Proof: The proof is straightforward, and thus it is omitted.

Suppose that s and t are two periods such that $1 \leq s \leq t \leq T$, and let us define more notation:

- D(s,t) denotes the demand with earliest due dates at s and latest due dates at t. We assume that D(s,t) is given.
- D_t denotes the sum of all demands with latest due dates at t.
- $A(s,t)$ denotes the sum of all demands with latest due dates within $[s, t]$.
- $B(s,t)$ denotes the sum of all demands with time windows located within $[s, t]$, and
- $E(s,t)$ denotes the sum of all demands with earliest due dates at s and latest due dates during $[s ,t]$.

By definition, $A(s,t) \geq B(s,t) \geq E(s,t)$. Observe that D_t and $A(s,t)$ can indeed be calculated in a straightforward way in $O(T^2)$ time. Also, note that $B(s,t) = B(s+1,t) + E(s,t)$. Hence, $B(s,t)$, $1 \leq s \leq t \leq T$, can be computed in (overall) $O(T^2)$ time if all the $E(s,t)$ values are known. To compute the the $E(s,t)$ values, we first have to sort the demands according to the values E_i and L_i. Using a bucket sort procedure, this can again be done in $O(T^2)$. Next, we can compute $E(s,t)$ using $E(s,t) = E(s,t\text{-}1) + D(s,t)$. That is, we can find all $E(s,t)$ in $O(T^2)$ time. In turn, we can find D_t, $A(s,t)$, $B(s,t)$, and $E(s,t)$ for all possible s and t in $O(T^2)$ time.

Now, we develop dynamic programming algorithms for the two different cases of the problem, i.e., **Case 1: Shortages are not Allowed**, and **Case 2: Backordering is Allowed**. As we will see shortly, Properties 1 and 2 are useful in simplifying the computational difficulty and decreasing the time complexity of our algorithms for both cases.

5. Case 1: Shortages are not Allowed

If shortages are not allowed then the following property is a direct result of Property 2.

Property 3: Suppose that $t_1 < t_2 < \ldots < t_r$ are the consecutive replenishment periods in an optimal solution. Then the replenishment order received in period t_j is used to satisfy those d_i with $t_j \leq L_i < t_{j+1}$.

Knowing that Property 3 holds, let $C(s,t)$ denote the minimum cost of buying/producing in period s to satisfy those d_i with $s \leq L_i \leq t$. Also, let $F(t)$ denote the minimum cost of satisfying those d_i with $L_i \leq t$. It follows that

$F(0) = 0$,

$$F(t) = \min\{F(s\text{-}1) + C(s,t): 1 \leq s \leq t\},\ t = 1,\ldots,T, \tag{10}$$

and the optimal value is given by $F(T)$.

Observe that if the $C(s,t)$ values have already been computed, then the computational complexity of computing $F(T)$ is $O(T^2)$. In this case, the remaining question is how to compute $C(s,t)$ efficiently. This can be achieved by using the following recursive iterations. For $t = T, T-1,\ldots,1$, let $q \leq t$ be the latest period with $L_i = q$ for some d_i. Then, $C(s,t) = 0$ for $q < s \leq t$ and

$$C(q,t) = K_q\,. \tag{11}$$

For $1 \leq s < q$, we have

$$C(s,t) = C(s+1,t) + K_s - K_{s+1} + h_s B(s+1,t). \tag{12}$$

Here is the justification of (12). Given that we already calculated $C(s+1,t)$, the minimum cost of replenishing stock in period $s+1$ to satisfy those d_i with $s+1 \leq L_i \leq t$, $C(s,t)$ should be the sum of

- $C(s+1,t)$,
- the incremental fixed cost of replenishing stock in period s, i.e., $K_s - K_{s+1}$, and
- the cost of holding those d_i with time windows located within $[s+1, t]$, i.e., $h_s B(s+1,t)$.

Observe that regardless whether s or $s+1$ is the period of replenishment for satisfying those d_i with $s+1 \leq L_i \leq t$, we have $I_r^+ = B(r+1,t)$ for $r = s+1,\ldots,t$. Furthermore, if s is the period of replenishment for satisfying d_i with $s \leq L_i \leq t$ then $I_s^+ = B(s+1,t)$ Thus, $C(s,t)$ should include the cost of holding $B(s+1,t)$ during s.

Recall that the computational complexity of calculating $B(s,t)$ for $1 \leq s \leq t \leq T$ is given by $O(T^2)$. Once the $B(s,t)$ values have been calculated, for a given t ($1 \leq t \leq T$) we can compute $C(s,t)$ for $1 \leq s \leq t$ in $O(T)$ time. Thus, the computational complexity of calculating $C(s,t)$ for $1 \leq s \leq t \leq T$ is again given by $O(T^2)$. It now follows that the computational complexity of our proposed algorithm is $O(T^2)$.

Remark 1: Recently, Federgruen and Tzur (1991), Wagelmans et al. (1992), and Aggarwal and Park (1993) have developed O(T) algorithms to solve the classical dynamic lot-sizing problems for which $n \leq T$. One of the key ideas in those papers is to transform the mathematical formulation of the classical dynamic lot-sizing model to an equivalent formulation without holding costs. Unfortunately, for the more general problem with demand time window considerations, equation (4) seems to make such a transformation difficult. Hence, development of an O(T) algorithm also seems to be difficult. However, this is not a real difficulty because, for the more general problem with demand time window considerations, n may be of order T^2. It follows that even reading the data takes up to $O(T^2)$ time. Therefore, an optimal algorithm with complexity $O(T^2)$ is in fact satisfactory.

6. Case 2: Backordering is Allowed

Now, we analyze the case where backordering is allowed. We note that, in the following, the definition of $C(s,t)$ is slightly different from Case 1. We also define additional notation.

- For $1 \leq s < t \leq T$, $C(s,t)$ denotes the minimum cost of replenishing in **periods *s* and *t*** to satisfy those d_i with $s < L_i \leq t$. Here, we intentionally do not include the cost of satisfying those d_i with $s = L_i$ in the definition since it will be considered in the calculation of $C(r,s)$ for some $r < s$.
- For $1 \leq t \leq T$, $\overline{C}(0,t)$ denotes the minimum cost of satisfying d_i with $0 < L_i \leq t$ where t is the *first* replenishment period. Note that, by definition $\overline{C}(0,t)$ includes a fixed replenishment cost even if there is no actual demand to satisfy.
- For $1 \leq s \leq T$, $\overline{\overline{C}}(s,T)$ denotes the minimum cost of satisfying d_i with $s < L_i \leq T$ where s is the *last* replenishment period.
- By definition, $F(0) = 0$. For $1 \leq t \leq T$-1, we define $F(t)$ as the minimum cost of satisfying those d_i with $L_i \leq t$ if t is the last period of replenishment.
- Also, we define $F^*(T)$ as the minimum cost of satisfying those d_i with $L_i \leq T$. Hence, $F^*(T)$ denotes the optimal cost.

It follows that, for $t = 1,\ldots,T-1$

$$F(t) = Min\begin{cases} \overline{C}(0,t) \\ Min\{F(s) + C(s,t) - K_S : s < t\} \end{cases}$$

Observe that, by definition, both $F(s)$ and $C(s,t)$ include the fixed cost of replenishing in period s. Thus, in order to obtain an expression of $F(t)$, we subtract K_s from $F(s) + C(s,t)$. Knowing $F(t)$ for t =1,…,T-1, the optimal solution is obtained through the following recursive equations:

$$F^*(T) = Min\begin{cases} Min\{F(s) + \overline{\overline{C}}(s,T) - K_S : s < T\} \\ Min\{F(s) + C(s,T) - K_S : s < T\} \end{cases} \tag{13}$$

Also, note that T may or may not be a replenishment period. Therefore, either one of the following two can happen: if $F^*(T)$ = $Min\{F(s)+\overline{\overline{C}}(s,T)-K_S : s<T\}$, then T is not a replenishment period in an optimal solution, and otherwise T is a replenishment period.

If $\overline{C}(0,t)$, $\overline{\overline{C}}(s,T)$ and $C(s,t)$, $1 \le s \le t \le T$ have already been calculated then the computational complexity of obtaining $F^*(T)$ is $O(T^2)$. Thus, the important question is how to compute $\overline{C}(0,t)$, $\overline{\overline{C}}(s,T)$ and $C(s,t)$ for $1 \le s \le t \le T$ effectively. In order to answer this question, in the remainder of this section we focus on providing efficient procedures for computing all possible $\overline{C}(0,t)$, $\overline{\overline{C}}(s,T)$ and $C(s,t)$ values. We show that those computations can be completed in $O(T^3)$ time, and hence the complexity of the proposed algorithm is $O(T^3)$.

Computing $\overline{C}(0,t)$

By definition, we simply have

$$\overline{C}(0,1) = K_1, \tag{14}$$

and

$$\overline{C}(0,t+1) = \overline{C}(0,t) + K_{t+1} - K_t + b_t A(1,t), \quad t = 1,2,\ldots,T-1. \tag{15}$$

We also have $I_r^- = A(1,r\text{-}1)$ for r = 1,…,t-1 regardless whether t or t+1 is the first period of replenishment for satisfying those d_i with $1 \le L_i \le t$. Furthermore, $I_t^- = A(1,t)$ if t+1 is the first period of replenishment for satisfying those d_i with $s \le L_i \le t+1$. Thus, the difference between $\overline{C}(0,t)$ and $\overline{C}(0,t+1)$ is given by the incremental fixed cost $K_{t+1} - K_t$, and the backordering cost $b_t A(1,t)$. Hence, equation (15) is true.

If $A(1,t)$, t =1,…,T, values have already been calculated then - given $\overline{C}(0,t)$ - $\overline{C}(0,t+1)$ can be computed in constant time. Thus, it takes O(T) time to calculate all $\overline{C}(0,t)$.

Computing $\overline{\overline{C}}(s,T)$

Observe that

$$\overline{\overline{C}}(T-1,T) = K_{T-1} + h_{T-1}B(T,T), \tag{16}$$

and

$$\overline{\overline{C}}(s,T) = \overline{\overline{C}}(s+1,T) + K_s - K_{s+1} + h_s B(s+1,T), \quad u = T\text{-}2, T\text{-}3, \ldots, 1. \tag{17}$$

The justification of (17) is similar to that of (15). If $B(s,T)$, s =1,…,T, values have already been calculated then, given $\overline{\overline{C}}(s+1,T)$, $\overline{\overline{C}}(s,T)$ can be computed in constant time, and thus, the computational complexity of obtaining all $\overline{\overline{C}}(s,T)$ is O(T).

Computing *C(s,t)*

Given two consecutive replenishment periods s and t, if $s < E_i \leq L_i < t$ then we need to know whether it is cheaper to satisfy d_i using the replenishment quantity received in s or in t. If $E_i = u$, and $L_i = v$, then the answer depends only on whether or not

$$h_s + h_{s+1} + \ldots + h_{u-1} > b_v + b_{v+1} + \ldots + b_{t-1}. \tag{18}$$

For each triple (u,v,t) with $1 < u \leq v \leq t$, let us define $q(u,v,t)$ as the latest period s < u for which inequality (18) is satisfied. Suppose that j and t are two consecutive replenishment periods such that $j < u \leq v \leq t$. If $j \leq q(u,v,t)$, then it is cheaper to satisfy d_i by using the replenishment received in period t. Otherwise, it is cheaper to satisfy d_i by using the replenishment received in period j. Because of monotonicity properties, we can compute all possible $q(u,v,t)$ in $O(T^3)$ time.

In order to find an efficient way for calculating $C(s,t)$, we need to derive a recursive relation between $C(s,t)$ and $C(s+1,t)$. In deriving a relationship between $C(s,t)$ and $C(s+1,t)$, we need to analyze the changes in three cost items: setup cost, holding cost, and backordering cost. That is, in deriving a recursive relation between $C(s,t)$ and $C(s+1,t)$, we need to know the quantities of the demands that were previously ordered in period $s+1$ (when $s+1$ and t are two consecutive replenishment periods) but now should be ordered in period s (when s and t become two consecutive replenishment periods). In order to keep track of this amount, we define the following:

- For u such that $s < u < t$, let S(s,u,t) be the set of all v with $q(u,v,t) = s$ where $u \le v < t$. Also let $\Delta_1(s,u,t) = \sum_{v \in S(s,u,t)} D(u,v)$. It follows that among B($s+2$, $t-1$) (i.e., those d_i with $s+2 \le E_i \le L_i \le t-1$), $\sum_{u=s+2}^{t-1} \Delta_1(s,u,t)$ units ordered in $s+1$ (when $s+1$ and t are two consecutive replenishment periods) are ordered in t when s and t become two consecutive replenishment periods.
- For v such that $s < v < t$, let T(s,v,t) be the set of all u with $q(u,v,t) = s$ where $s < u \le v$. Also, let $\Delta_2(s,v,t) = \sum_{u \in T(s,v,t)} D(u,v)$. It follows that among B($s+2$, $t-1$), $\sum_{u=s+2}^{t-1} \Delta_2(s,v,t)$ units ordered in $s+1$ (when $s+1$ and t are two consecutive replenishment periods) are ordered in t when s and t become two consecutive replenishment periods. Thus, we have

$$\sum_{u=s+2}^{t-1} \Delta_1(s,u,t) = \sum_{v=s+2}^{t-1} \Delta_2(s,v,t).$$

It is worth noting that $\Delta_1(s,u,t)$ and $\Delta_2(s,v,t)$ are useful in deriving a recursive relationship between $C(s,t)$ and $C(s+1,t)$. That is, $\Delta_1(s,u,t)$ is used for calculating the holding cost decrease when s and t become two consecutive replenishment periods instead of $s+1$ and t. Similarly, $\Delta_2(s,v,t)$ is used for calculating the backordering cost increase .

Given $q(u,v,t)$ values, we can calculate $\Delta_1(s,u,t)$ values as follows: Initialize with all $\Delta_1(s,u,t)$ set to zero. Consider the values $q(u,v,t)$ one by one and add $D(u,v)$ to $\Delta_1(s,u,t)$ if $q(u,v,t) = s$. Using this method, it takes $O(T^3)$ time to calculate all $\Delta_1(s,u,t)$. Adopting a similar method, the computational complexity of obtaining the $\Delta_2(s,v,t)$ values is also $O(T^3)$.

Let $I(s,t)$ denote the optimal level of on-hand inventory at the end of period s (i.e. $I(s,t) = I_s^+$), given that s and t are two consecutive replenishment periods for satisfying those d_i with $s+1 \le L_i \le t$. The $I(s,t)$ values can be computed using the following backward iterations:

$$I(s,t) = I(s+1,t) - \sum_{u=s+2}^{t-1} \Delta_1(s,u,t) + E(s+1,t-1) - \Delta_1(s,s+1,t)$$

$$= I(s+1,t) - \sum_{u=s+1}^{t-1} \Delta_1(s,u,t) + E(s+1,t)$$

Regardless that "s and t" or "s+1 and t" are consecutive replenishment periods, it is always cheaper to satisfy those d_i with $L_i = t$ by the replenishment in period t (i.e., without incurring any holding or backordering costs). The above recursion formula reflects the fact that, if s and t are two consecutive replenishment periods then it is cheaper to order for the $\sum_{u=s+2}^{t-1} \Delta_1(s,u,t)$ units in t. Furthermore, when s and t become two consecutive replenishment periods then $E(s+1,t-1)$ (i.e., all demands with earliest due dates in s+1 and latest due date before t) should be carried in inventory at the end of period s, except for the amount $\Delta_1(s,s+1,t)$ (i.e., the part that can be replenished at a lower cost in t instead of s).

Now, for $1 \le s \le t < T$, let

$h(s,t) = h_s + h_{s+1} + \ldots + h_t$,

and

$b(s,t) = b_s + b_{s+1} + \ldots + b_t$.

Thus, for $t = 2,\ldots,T$, we have

$C(t\text{-}1, t) = K_{t-1} + K_t$, (19)

and for $s = t$-1,t-2,…,1, we have

$$C(s,t) = C(s+1, t) + K_s - K_{s+1}$$

$$- \sum_{u=s+2}^{t-1} h(s+1,u-1)\Delta_1(s,u,t) + \sum_{v=s+2}^{t-1} b(v,t-1)\Delta_2(s,u,t)$$

$$+ h_s I(s,t)$$

$$+ b(s+1,t-1)\Delta_2(s,s+1,t) \quad (20)$$

The correctness of the above formula follows from the following observations:

i) As we have mentioned earlier, among B(s+2,t-1), $\sum_{u=s+2}^{t-1} \Delta_1(s,u,t)$ units that are replenished in s+1 when s+1 and t are two consecutive replenishment periods, should be replenished in period t when s and t becomes two consecutive replenishment periods. Therefore, in computing $C(s,t)$ we subtract the corresponding holding cost of $\sum_{u=s+2}^{t-1} h(s+1,u-1)\Delta_1(s,u,t)$ from $C(s+1, t)$, but we do add a backordering cost of $\sum_{v=s+2}^{t-1} b(v,t-1)\Delta_2(s,u,t)$. Recall that $\sum_{u=s+2}^{t-1} \Delta_1(s,u,t)$ = $\sum_{u=s+2}^{t-1} \Delta_2(s,v,t)$.

ii) $I(s,t)$ is the on hand inventory at the end of period s when s and t are two consecutive replenishment periods, and hence we include a holding cost of $h_s I(s,t)$ in calculating $C(s,t)$.

iii) If s+1 and t are two consecutive replenishment periods, then it is cheaper to satisfy $E(s+1, t-1)$ (i.e., those d_i with $E_i = s+1$, and $s+1 \le L_i \le t-1$) using the replenishment received in s+1 (without incurring any holding or backordering costs). However, among $E(s+1, t-1)$, $\Delta_2(s,s+1,t)$ is the amount to be produced in period t when s and t become two consecutive replenishment periods. Therefore, in computing $C(s,t)$, we consider a backordering cost of $b(s+1,t-1)\Delta_2(s,s+1,v)$.

Since it takes $O(T^3)$ to find all $\Delta_1(s,u,t)$ and $\Delta_2(s,v,t)$ values, it also takes $O(T^3)$ to compute $C(s,t)$ for $1 \leq s \leq t \leq T$. Hence, the complexity of the algorithm for computing $F(T)$ is $O(T^3)$.

7. Extension to General Case

In this section, we generalize our results to the case $p_t \geq p_{t+1}$ and $p_t + b_t \geq p_{t-1}$ for all t. Observe that $p_t \geq p_{t+1}$ implies $p_t + h_t \geq p_{t+1}$, and thus our analysis concentrates on a case where there are no speculative motives to hold inventory or to backorder. Under these assumptions, the algorithms developed in Section 5 and 6 can be generalized as illustrated below. Since the unit ordering/procurement costs are no longer stationary, we now consider a time dependent ordering/procurement cost function in our recursive calculations. Thus, we replace equations (11) – (12) and (14) - (20) by the following equations.

$$C(q,t) = K_q + p_q D_q \qquad (11)'$$

$$C(s,t) = C(s+1,t) + K_s - K_{s+1} + (p_s - p_{s+1})A(s+1, t) + p_s D_s + h_s B(s+1,t). \qquad (12)'$$

$$\overline{C}(0,1) = K_1 + p_1 D_1 \qquad (14)'$$

$$\overline{C}(0,t+1) = \overline{C}(0,t) + K_{t+1} - K_t + (p_{t+1} - p_t)A(1,t) + p_{t+1}D_{t+1} + b_t A(1,t) \qquad (15)'$$

$$\overline{\overline{C}}(T-1,T) = K_{T-1} + p_{T-1}A(T,T) + h_{T-1}B(T,T) \qquad (16)'$$

$$\overline{\overline{C}}(s,T) = \overline{\overline{C}}(s+1,T) + K_s - K_{s+1} + (p_s - p_{s+1})A(s+2, T) + p_s D_{s+1} + h_s B(s+1,T) \qquad (17)'$$

$$p_s + h_s + h_{s+1} + \ldots + h_{u-1} > b_v + b_{v+1} + \ldots + b_{t-1} + p_t \qquad (18)'$$

$$C(t-1, t) = K_{t-1} + K_t + p_t A(t,t) \qquad (19)'$$

$$\begin{aligned} C(s,t) = {} & C(s+1, t) + K_s - K_{s+1} \\ & - \sum_{u=s+2}^{t-1} h(s+1,u-1)\Delta_1(s,u,t) + \sum_{v=s+2}^{t-1} b(v,t-1)\Delta_2(s,v,t) + (p_t - p_{s+1}) \sum_{u=s+2}^{t-1} \Delta_1(s,u,t) \\ & + h_s I(s,t) \\ & + p_s (A(s+1,t)-B(s+1,t-1)) - p_{s+1}(A(s+2,t)-B(s+2,t-1)). \\ & + b(s+1,t-1)\Delta_2(s,s+1,t) + p_t \Delta_2(s,s+1,t) \end{aligned} \qquad (20)'$$

Based on the cost assumptions of the problem of interest, equations (11)′-(12)′ and (14)′-(19)′ are straightforward to justify. The following observations provide a justification for equation (20)′:

i) As we have mentioned earlier, if s and t are two consecutive replenishment periods, $\sum_{u=s+2}^{t-1} \Delta_1(s,u,t)$ units ordered in s+1 previously (when s+1 and t are two consecutive replenishment periods) are ordered in t. Therefore, in computing $C(s,t)$, we consider the incremental cost of ordering/buying $\sum_{u=s+2}^{t-1} \Delta_1(s,u,t)$ in period t. This cost is given by $(p_t - p_{s+1}) \sum_{u=s+2}^{t-1} \Delta_1(s,u,t)$.

ii) $A(s+2,t-1) - B(s+2,t-1)$ is the sum of those d_i with $E_i \leq s+1$ and $s+2 \leq L_i \leq t-1$. Clearly, if s+1 and t are two consecutive replenishment periods, then it is cheaper to satisfy $A(s+2,t-1) - B(s+2,t-1)$ by using the replenishment quantity in received in s+1. In this case, $p_{s+1}[A(s+2,t)-B(s+2,t-1)]$ represents the corresponding ordering/procurement cost (excluding the fixed cost). Similarly, $A(s+1,t-1)-B(s+1,t-1)$ is the sum of those d_i with $E_i \leq s$ and $s+1 \leq L_i \leq t-1$. If s and t are two consecutive replenishment periods, it is cheaper to satisfy this quantity through a replenishment decision in s, and the corresponding ordering/procurement cost is $p_s[A(s+1,t)-B(s+1,t-1)]$ (excluding the fixed cost).

iii) Recall that, if s+1 and t are two consecutive replenishment periods then it is cheaper to satisfy $E(s+1,t-1)$ (i.e., those d_i with $E_i = s+1$, and $s+1 \leq L_i \leq t-1$) using the replenishment quantity received in s+1 (without incurring any hold or backlogging costs). However, among $E(s+1,t-1)$, $\Delta_2(s,s+1,t)$ is the amount to be replenished in period t when s and t become two consecutive replenishment periods. The cost of replenishing $\Delta_2(s,s+1,t)$ units (excluding the fixed cost) in s+1 and t are $p_{s+1}\Delta_2(s,s+1,t)$ and $p_t\Delta_2(s,s+1,t)$, respectively. Note that $p_{s+1}\Delta_2(s,s+1,t)$ is already included in $p_{s+1}[A(s+2,t)-B(s+2,t-1)]$. Hence, we need to include $p_t\Delta_2(s,s+1,t)$ in Equation (20)′.

It is left to the reader to verify that the running time of the algorithm is still $O(T^3)$ in the backlogging case and $O(T^2)$ if backlogging is not allowed. Finally, we note that for the backlogging case, the time bound can also be achieved when $p_t+h_t \geq p_{t+1} \geq p_t$ for all t, because reversing the time axis this case can be transformed into the case discussed above.

8. Another Application Area and Future Research

If the unit holding and backordering cost is very high, or equivalently, ordering before the earliest due date or after the latest due date is not allowed, then each demand needs to be ordered within its demand time window. It is also interesting to observe that this specific case of our problem is equivalent to the following location problem. "What will be the minimum cost of installing facilities that can serve all one-dimensional customers where customer i is located at $(E_i + L_i)/2$ and can only be served by a facility within a distance $(L_i - E_i)/2$?" (Francis et al., 1992) Therefore, the dynamic programming algorithm described above is also applicable in the context of location planning.

Some important research directions in dynamic lot-sizing with demand time windows include consideration of capacity constraints and the incorporation of inbound and outbound transportation costs.

References

Aggarwal, A. and J. K. Park, "Improved Algorithms for Economic Lot-Size Problems," *Operations Research*, 41, (1993), pp. 549-571.

Anger, F. D., C.-Y. Lee, and L. A. Martin-Vega, "Single Machine Scheduling with Tight Windows," Research Report No. 86-16, Industrial and Systems Engineering Department, University of Florida, 1986.

Aviv, Y. and A. Federgruen, "The Operational Benefits of Information Sharing and Vendor Managed Inventory Programs," Technical Report, Industrial Engineering and Operations Research Department, Columbia University, 1998.

Bramel, J. and D. Simchi-Levi, *The Logic of Logistics*, (1997), Springer, New York, N.Y.

Bitran, G. R., T. L. Magnanti, and H. H. Yanasse, "Approximation Methods for the Uncapacitated Dynamic Lot-Size Problem," *Management Science*, 30, (1984), pp. 1121-1140.

Çetinkaya, S. and C.-Y. Lee, "Stock Replenishment and Shipment Scheduling for Vendor Managed Inventory Systems," Technical Report, Industrial Engineering

Department, Texas A&M University, 1998 (to appear in *Management Science*, accepted July 1999).
Çetinkaya, S. and C.-Y. Lee, "Static Demand Models for Coordinating Inventory and Outbound Transportation Decisions at a Third Party Warehouse," Technical Report, Industrial Engineering Department, Texas A&M University, 1999.
Chen, H.-D. and C.-Y. Lee, "A Simple Algorithm for the Error Bound of the Dynamic Lot-Size Model Allowing Speculative Motive," *IIE Transactions*, 27, (1995), pp. 683-388.
Chen, H.-D., D. W. Hearn, and C.-Y. Lee, "A New Dynamic Programming Algorithm for the Single Item Capacitated Dynamic Lot-Size Model," *Journal of Global Optimization*, 4, (1994), pp. 285-300.
Cheng, T. C. E., "Optimal Common Due Date with Limited Completion Time Deviation," *Computers and Operations Research*, 15 (1988), pp. 91-96.
Chung, C. S. and C. H. M. Lin, "An $O(T^2)$ Algorithm for the NI/G/NI/ND Capaciated Lot-Size Problem," *Management Science,* 34, (1988), pp. 420-426.
Desrochers, M., J. Desrosiers, and M. Solomon, "A New Optimization Algorithm for the Vehicle Routing Problem with Time Windows," *Operations Research*, 40, (1992), pp. 342-254.
Evans, J.R., "An Efficient Implementation of the Wagner-Whitin Algorithm for Dynamic Lot-Sizing," *Journal of Operations Management,* 5, (1985), pp. 229-235.
Federgruen, A. and M. Tzur, "A Simple Forward Algorithm to Solve General Dynamic Lot-Sizing Models with n Periods in O(n log n) or O(n) Time," *Management Science,* 37, (1991), pp. 909-925.
Fisher, M. L., K. O. Jornsten, and O. B. G. Madsen, "Vehicle Routing with Time Windows: Two Optimization Algorithms," *Operations Research*, 45, (1997), pp. 488-492.
Florian, M., J. K. Lenstra, and A. H. G. Rinnooy Kan, "Deterministic Production Planning Algorithm and Complexity," *Management Science*, 26, (1980), pp. 669-679.
Francis, R.L., L. F. McGinnis and J.A. White, *Facility Layout and Location: An analytical Approach,* second edition, Prentice-Hall, Inc., Englewood Cliffs, N.J. 1992.
Gaither, N., "An Improved Lot-Sizing Model for MRP Systems," *Production and Inventory Management,* 24-3, (1983), pp. 10-19.
Johnson, L. A. and D. C. Montgomery, *Operations Research in Production Planning, Scheduling, and Inventory Control*, (1974), John Wiley and Sons, New York, N.Y.
Kohl, N. and O. B. G. Madsen, "An Optimization Algorithm for the Vehicle Routing Problem with Time Windows Based on Largangian Relaxation," *Operations Research*, 45, (1997), pp. 395-406.
Kohl, N., J. Desrosiers, O. B. J. Madsen, M. M. Solomon, and F. Soumis, "k-Path Cuts for the Vehicle Routing Problem with Time Windows," (1998), to appear in *Transportation Science.*
Kraemer, F. and C.-Y. Lee, "Common Due Window Scheduling," *Production and Operations Management*, 2, (1993), pp. 262-275.

Kraemer, F. and C.-Y. Lee, “Due Window Scheduling for Parallel Machines,” *Mathematical and Computer Modelling, Special Issue on Scheduling: Theory and Applications*, 20, (1994), pp. 69-89.

Lee, C.-Y. and E. V. Denardo, “Rolling Planning Horizon: Error Bounds for the Dynamic Lot-Size Model,” *Mathematics of Operations Research*, 11, (1986), pp. 423-432.

Lee, C.-Y., “A Solution to the Multiple Set-Up Problem with Dynamic Demand,” *IIE Transactions,* 21, (1989), pp. 266-270.

Liman, S. D. and S. Ranaswamy, “Earliness-Tardiness Scheduling Problem with a Common Delivery Window,” *Operations Research Letters*, 15, (1994), pp. 195-203.

Liman, S. D., S. S. Panwalkar, and S. Thongmee, “Determination of Common Due Window Location in a Single Machine Scheduling Problem,” *European Journal of Operational Research*, 93, (1996), pp. 68-74.

Maes, J. and L. N. Van Wassenhove, “A Simple Heurisitc for the Multi-Item Single Level Capacitated Lot-Sizing Problem,” *Operations Research Letters,* 4, (1985), pp. 265-273.

Parker, K., “Demand Management and Beyond,” *Manufacturing Systems*, June, (1996), pp. 2A-14A.

Shaw, D. X. and A. P. M. Wagelmans, “An Algorithm for Single-Item Capacitated Economic Lot-Sizing with Piecewise Linear Production Costs and General Holding Costs,” *Management Science,* 44, (1998), pp. 831-838.

Silver, E. A. and H. E. Meal, “A Heuristic for Selecting Lot-Size Quantities for the Case of a Deterministic Time Varying Demand Rate and Discrete Opportunities for Replenishment,” *Production and Inventory Management*, 26, (1980), pp. 669-679.

Silver, E. A., D. F. Pyke, and R. Peterson, *Inventory Management and Production Planning and Scheduling, 3rd Edition,* (1998), John Wiley & Sons, New York, N.Y.

Solomon, M. M. and J. Desrosiers, “Time Window Constrained Routing and Scheduling Problems: A Survey,” *Transportation Science*, 22, (1988), pp. 1-13.

Swoveland, C. “A Deterministic Multi-Period Production Planning Model with Piece-wise Concave Production and Holding-Backlogging Costs,” *Management Science*, 21, (1975), pp. 1007-1013.

Van Hoesel, C. P. M. and A. P. M. Wagelmans, “An O(T^3) Algorithm for the Economic Lot-Sizing Problem with Constant Capacities,” *Management Science*, 42, (1996), pp. 142-150.

Van Hoesel, C. P. M. and A. P. M. Wagelmans, “Fully Polynomial Approximation Schemes for Single-Item Capacitated Economic Lot-Sizing Problems,” Report No. 9735/A, Erasmus University Rotterdam, 1997.

M. X. Ventura and J. A. Weng, “Scheduling about a large Common Due Date with Tolerance to Minimize Mean Absolute Deviation of Completion Times,” *Naval Research Logistics,* 41, (1994), pp. 843-851.

M. X. Weng and J. A. Ventura, “A Note on Common Due Window Scheduling” *Production and Operations Management,* 5, (1996), pp. 194-200.

Wagelmans, A. P. M., S. Van Hoesel, and A. Kolen, “Economic Lot-Sizing: an O(nlogn)-Algorithm that Runs in Linear Time in the Wagner-Whitin Case,” *Operations Research,* 40, (1992), pp. S145-S156.

Wagner, H. M. and T. M. Whitin, “Dynamic Version of the Economic Lot-Size Model,” *Mangement Science*, 5-1, (1958), pp. 89-96.

Zangwill, W. I., “A Deterministic Multi-Period Production Scheduling Model with Backlogging,” *Management Science*, 13, (1966), pp. 105-119.